

# Linux Tutorium

## 8. Netzwerk - Grundlagen

# Netzwerk - Grundlagen

- Hostname

- einem Rechner, der sich im Netzwerk befindet, gibt man einen so genannten Hostnamen – einen Namen, der im jeweiligen Netzwerk einzigartig ist und zur Identifikation des Rechners (Hosts) dient
- man unterscheidet dabei zwischen einem bloßen Hostnamen und einem FQDN (Fully Qualified Domain Name)
  - bei dem Hostnamen handelt es sich um einen einfachen Namen, z.B. „karl“
  - FQDN besteht aus dem einfachen Hostname und der kompletten Domain, der dieser Host angehört z.B. „karl.sun.informatik“

# Netzwerk - Grundlagen

– es wird der Befehl „hostname“ benötigt

- um Hosts in Netzwerken zu plazieren
- um Hosts einen Namen zu geben

```
$ hostname karl.sun.informatik
```

- um den aktuellen Namen anzulesen

```
$ hostname
```

```
karl
```

```
$ hostname -f
```

```
karl.sun.informatik
```

# Netzwerk - Grundlagen

- Netzwerkkarten
    - unter Linux erhalten Netzwerkkarten die Bezeichnung ethX, wobei X eine laufende Nummer darstellt, die mit jeder Schnittstelle erhöht wird (Die erste Schnittstelle heißt eth0, die zweite eth1, usw.)
    - die Administration von Netzwerkschnittstellen kann via „ifconfig“ erfolgen
      - wichtige Funktionen von ifconfig sind u.a.
        - Auflistung aller im System integrierten Netzwerkschnittstellen (am besten den Parameter -a übergeben, um wirklich alle Netzwerkschnittstellen anzuzeigen)
- ```
# ifconfig -a
```

# Netzwerk - Grundlagen

- IP-Adresse setzen (um eine Adresse zu setzen, übergibt man ifconfig den Namen der gewünschten Schnittstelle sowie die zugehörige IP-Adresse)

```
# ifconfig eth0 10.10.10.1
```

- Netzmaske setzen

```
# ifconfig eth0 192.168.0.2 netmask 255.255.255.0
```

- die Konfiguration mit ifconfig bleibt nur während der Laufzeit des Systems bestehen
- nach einem Neustart müssen die Netzwerkschnittstellen neu konfiguriert werden
- eine andere Möglichkeit, eine Netzwerkkarte ganz einfach zu konfigurieren, ist DHCP (Dynamic Host Configuration Protocol)

# Netzwerk - Grundlagen

- der Vorteil, hierbei braucht man Allgemeinen gar nichts von Hand einzustellen – das geht automatisch
- der Nachteil ist natürlich, dass ein DHCP-Server im Netzwerk benötigt wird
- dieser Server verteilt an anfragende Hosts automatisch eine freie (oder je nach Konfiguration auch eine ganz bestimmte) IP-Adresse, und er teilt diesem Rechner mit, welches Default-Gateway er benutzen soll
- die meisten Distributionen lassen bei der Installation die Wahl zwischen dem automatischen Beziehen oder der statischen Methode

# Netzwerk - Grundlagen

- für den Fall, dass die Schnittstelle nicht statisch per Hand, sondern mit DHCP konfigurieren werden soll, sei kurz auf das Programm „dhcp-client“ und dessen Äquivalent „dhclient“ hingewiesen
- über einen einfachen Aufruf kann die Netzwerkschnittstelle per DHCP konfiguriert werden (hier am Beispiel der Schnittstelle eth1)

```
# dhcp-client -i eth1
```

```
// Alternativ:
```

```
# dhclient eth1
```

# Netzwerk - Grundlagen

- Wireless LAN
  - es braucht zwei Sachen um unter Linux auch mit WLAN arbeiten zu können
    - den richtigen WLAN-Treiber
    - die „wireless-tools“
  - WLAN Einstellungen legt man nicht mittels ifconfig, sondern mit dem „iwconfig“ Tool fest
  - wichtige Funktionen von iwconfig sind u.a.
    - ESSID setzen
    - Channel setzen
    - Netzwerk-ID setzen
    - ...



# Netzwerk - Grundlagen

- um verfügbare drahtlose Netzwerke zu finden benötigt man einen weiteren Befehl aus dem wireless-tools-Paket welcher sich „iwlist“ nennt
- iwlist mit dem Parameter „scan(ning)“ erledigt diese Aufgabe für uns

```
# iwlist scanning
eth1 Scan completed :
      Cell 01 - Address: 00:0B:6B:30:15:C0
      ESSID:"zuhause"
      Protocol:IEEE 802.11bg
      Channel:1
      Encryption key:off
      Bit Rate:54 Mb/s
```

# Netzwerk - Grundlagen

- wie ifconfig und iwconfig bietet auch iwlist darüber hinaus noch weit mehr Funktionen an
- /etc/hosts
  - um die Kommunikation zwischen den Hosts im Netzwerk zu erleichtern, spricht man die anderen Systeme über deren Hostnamen und nicht über deren IP-Adressen an
  - jeder Unix/Linux-Rechner verfügt über die Datei /etc/hosts
  - in diese kann eine (kleine) Liste der vorhandenen Hostnamen inklusive deren IP-Adressen eingetragen werden

# Netzwerk - Grundlagen

- er Host kann über diese Datei schnell die zu einem Host gehörige IP-Adresse herausfinden
  - der Aufbau der Datei gestaltet sich denkbar einfach
    - zuerst wird die Adresse angegeben (entweder IPv4 oder IPv6), dahinter werden die dafür gültigen Hostnamen geschrieben
- ```
::1 localhost.sun.informatik localhost  
127.0.0.1 localhost.sun.informatik localhost  
192.168.0.4 gerd.sun.informatik gerd  
192.168.0.5 horst.sun.informatik horst
```
- diese Datei ist allerdings fast nur für kleine Netzwerke ohne DNS-Server nützlich

# Netzwerk - Grundlagen

- /etc/networks

- in der /etc/networks wird, ähnlich wie in der Datei hosts, eine Liste von Adressen abgelegt
- hier werden allerdings ganze Netzwerkadressen in Netzwerknamen übersetzt, was in Verbindung mit DNS angewandt werden kann
  - In der ersten Spalte steht offizielle Netzwerkname, in der zweiten die Netzwerkadresse (Spalte drei, wenn vorhanden, legt einen Alias fest)

```
linux$ cat /etc/networks
default          0.0.0.0
loopback         127.0.0.0
link-local       165.254.0.0
```

# Netzwerk - Grundlagen

- `/etc/resolv.conf`
  - über die Datei `/etc/resolv.conf` wird der lokale Resolver konfiguriert
  - ein Resolver ist dafür zuständig, DNS-Anfragen aufzulösen, d.h. dass der Resolver sich darum kümmert, dem System die DNS-Informationen zu beschaffen, die benötigt werden
  - dabei kann es sich beispielsweise um eine IP-Adresse für einen Hostnamen, einen Mail-Exchanger oder Ähnliches handeln
  - eine typische `resolv.conf` sieht in etwa folgendermaßen aus:

# Netzwerk - Grundlagen

```
linux$ cat /etc/resolv.conf  
nameserver 194.94.30.50  
nameserver 212.201.65.167
```

- das Keyword „**nameserver**“ legt einen Nameserver fest
- es können (je nach Implementierung) maximal 3 Nameserver angegeben werden, die in ihrer aufgelisteten Reihenfolge vom Resolver abgefragt werden, der wichtigste Server sollte immer an erster Stelle stehen
- weitere Keywords sind u.a. **domain**, **bind**, **lookup**, **search**, ...

# Linux Tutorium

## 9. Netzwerkdienste

# Netzwerkdienste

- Standarddienste

- „echo“ (UDP/TCP-Port 7)

- echo funktioniert nach einem sehr einfachen Prinzip: Der Client sendet einen Zeichen-String an den Server, und der Server antwortet darauf, indem er genau denselben Zeichenstring wieder zurück an den Client sendet

- „discard“ (UDP/TCP-Port 9)

- der Client verbindet sich mit einem Server, worauf der Server eine Gegenverbindung zum Client aufbaut
    - nachdem eine Antwort des Clients eingetroffen ist (und diese vom Server verworfen wurde), beendet sich der Dienst
    - der Nutzen besteht darin, Testverbindungen aufzubauen



# Netzwerkdienste

- „netstat“ (TCP-Port 15)
  - „netstat“ liefert Informationen über die Netzwerkverbindungen und Sockets des Systems (lokaler Portscanner)
    - a = Alle Ports
    - p = Programm, das den Port geöffnet hat, anzeigen lassen
    - t = TCP-Ports
    - u = UDP-Ports
- „finger“ (TCP-Port 79)
  - gibt Informationen über eingeloggte Benutzer an den Client weiter
  - der finger-Dienst kann u.a. verwendet werden, um zu überprüfen, ob ein Account existiert, ob er derzeit verwendet wird, wie lange der Benutzer bereits eingeloggt ist, ...

# Netzwerkdienste

```
$ finger linux
```

```
Login: linux          Name: Linus Torvalds
```

```
Directory: /home/linux Shell: /bin/bash
```

```
On since Fri Jun 26 17:06 (CET) on tty1
```

```
No mail.
```

- der finger-Dienst wird oft auch für andere Zwecke, nämlich zur Informationspräsentierung, verwendet

## – „telnet“

- telnet ist eine Terminalemulation über Netzwerk
- startet man den telnet-Client, erscheint ein telnet>-Prompt
- telnet ist unsicher, wenn SSL nicht aktiviert ist (was leider die Standardeinstellung ist)
- statt telnet sollte man besser SSH/OpenSSH benutzen

# Netzwerkdienste

## – „wget“

- möchte man gezielt eine Datei von einem Web- oder FTP-Server herunterladen, so genügt meistens die Features des Tools wget
- die einzige Voraussetzung für die Benutzung von wget ist, dass die exakte Adresse (URL) der Datei bekannt sein muß

```
$ wget http://www.beispiel.de/test.tar.gz
```

# Netzwerkdienste

- Network File System(NFS)
  - NFS ermöglicht es, Verzeichnisse – und das können in Form von Mountpoints auch ganze Partitionen sein – im Netzwerk bereitzustellen
  - Freigaben liegen physikalisch auf dem NFS-Server
  - NFS-Clients können über entsprechende Software (in der Regel bereits im Kernel implementiert) auf diese Freigaben zugreifen und sie einfach ins eigene Dateisystem einhängen
  - mit dem NFS kann praktisch jedes Unix-System, BSD-System und jedes Linux-System arbeiten
  - von SUN Microsystems entwickelt

# Netzwerkdienste

- wie bei Client/Server-Architekturen üblich, besteht der Nachteil natürlich darin, dass bei einem Serverausfall keiner mehr an seine Daten herankommt
- NFS ist nicht geeignet um auch Windows-PCs mit diesen Netzwerkfreigaben verbinden zu können
- Das NFS ist eine pure und saubere Unix/Linux-Lösung

# Netzwerkdienste

- Samba
  - Samba ist ein Datei-, Druck- und Domänen-Anmelde-Server unter Linux für die Anbindung von Windows-Clients
  - das entsprechend benötigte Protokoll heißt SMB welches auf NETBIOS aufbaut
  - Samba ist die Linux-Software, die das SMB-Protokoll versteht und implementiert
  - Samba ist sehr komplex, deshalb wird nur oberflächlich darauf eingegangen
  - weiter Informationen gibt es [hier](#)

# Netzwerkdienste

## – Windows-Freigaben mounten

- will man selbst Dateien freigeben, spielt man den Server
- will man als Client auf fremde Freigaben zugreifen, braucht man ebenfalls dieses Programmpaket
- da sich alles in das Gesamtkonzept von Linux einfügt, braucht man, um Freigaben einzubinden, nur ein spezielles mount-Kommando

```
# mount -t smbfs -o username=wilson,password=binford  
//Server/test /mnt/win
```

- in diesem Beispiel würde vom Rechner mit der Bezeichnung „Server“ die Freigabe test nach /mnt/win gemountet

- Alternativ kann auch der Befehl „smbmount“ (smbmnt) verwendet werden

```
# smbmount //Server/test /mnt/win -o username=wilson  
%binford,ip=192.168.0.2,workgroup=WORKGROUP
```

# Netzwerkdienste

- soll die Freigabe über einen längeren Zeitraum bzw. ständig bestehen ist es bequemer einen Eintrag in die `/etc/fstab` vorzunehmen

## – Windows-Freigaben aushängen

```
# umount /mnt/win
```

- oder per „`smbmount {Mountpunkt}`“

```
# smbmount /mnt/win
```

## – „`smbclient`“

- ist ein ftp-ähnlicher Kommandozeilen-Client für den Zugriff auf SMB-Ressourcen
- zu den Operationen gehören Dinge wie das Holen von Dateien vom Server auf den lokalen Rechner, Platzieren von Dateien vom lokalen Rechner auf den Server, etc.



# Netzwerkdienste

```
# smbclient -U wilson //Server/test
```

- um die verfügbaren Freigaben eines Servers anzuzeigen

```
# smbclient -U wilson -L //Server
```

- für die Arbeit mit dem Client sind folgende Befehle wichtig
  - cd
  - get
  - put
  - print für Druckerfreigaben
- möchte man selbst Dateien freigeben, müssen zunächst der Samba-Daemon „smbd“ sowie der „nmbd“, eine Art DNS für NETBIOS, laufen
- eine eigene Konfiguration ist notwendig, um den Dienst sinnvoll nutzen zu können (*smb.conf*)

# Netzwerkdienste

- FTP-Dienst
  - viele Systeme verfügen über einen vorinstallierten FTP-Server, den man nur in der `(x)inetd.conf` freischalten muss
  - Details zu Clientsoftware in Kapitel 11
- R-Tools
  - R-Tools stellen eine Telnet-ähnliche Funktionalität zur Verfügung, verhindern jedoch, dass man sich extra an einem System anmelden muss
  - da diese Dienste unsicher sind werden sie von vielen Systemen gar nicht mehr standardmäßig mit installiert
    - zu diesen Tools zählen u.a. **rlogin,rcp,rexec,...**

# Linux Tutorium

## 10. Netzwerkdiagnose

# Netzwerkdiagnose

- für viele dieser Kommandos sind Superuser-Rechte erforderlich
- Kontrolle der Konfiguration
  - ifconfig (siehe Kapitel 8)
  - iwconfig (siehe Kapitel 8)
- Prüfen ob Host in IP-Netzwerk erreichbar ist und welche Zeit das Routing zu diesem in Anspruch nimmt (ICMP-Pakete)
  - „ping“  

```
$ ping -c4 debian.org
```

# Netzwerkdiagnose

- Routing-Informationen abrufen und ausgeben lassen
  - „route“ oder „netstat -r“
- alle Stationen auf dem Weg eines IP-Pakets anzeigen lassen
  - „traceroute“  
# traceroute debian.org
- IP-Adresse oder Domäne eines bestimmten Computers mittels DNS abfragen
  - „host“ oder „nslookup“  
# nslookup debian.org

# Netzwerkdiagnose

- mehr als nur ein Portscanner

- „nmap“

- # nmap debian.org

- muss manuell nach installiert werden
      - primäre Aufgabe ist mehr als eine IP zu scannen (u.a. ganze Adressbereiche z.B. 192.\*.0.1-200)
      - sehr großer Funktionsumfang der durch Parameter gesteuert wird

- Paketanalyse

- „tcpdump“ und „Wireshark“

- tcpdump horcht alle IP-Pakete ab, die auf ein Netzwerkgerät ankommen

# Netzwerkdiagnose

- unterstützt außerdem auch UDP, ICMP, ARP u.v.m.
- Wireshark kann tcpdump-Paketmitschnitte lesen, bietet somit eine bequeme Protokollanalyse
- Wireshark im Detail siehe Kapitel 11
- Fehlerursachen bei Netzwerkproblemen
  - Firewall blockt zuviel/falsch
  - schon benutzte MAC,IP oder Port belegt
  - falsche Netzkonfiguration bzw. für das falsche Netz konfiguriert
  - fehlender Gateway oder Proxy
  - Hardware-Defekt

# Linux Tutorium

## 11. Anwendungsszenarien



# Anwendungsszenarien

Wireshark

# Anwendungsszenarien

The screenshot shows a Debian virtual machine running in Sun xVM VirtualBox. The desktop environment includes a terminal window and the Wireshark application.

**1.** The terminal window shows the following commands and output:

```
debian:/home/linux# wireshark &
[1] 7512
debian:/home/linux# ping debian.org
PING debian.org (192.25.206.10) 56(84) bytes of data.
```

**2.** The Wireshark application is open, and the 'Capture Interfaces' dialog box is displayed. The 'eth1' interface is highlighted in the list.

Device	Description	IP	Packets	Packets/s	Start	Options
eth1		10.0.2.15	8016	129	Start	Options
any	Pseudo-device that captures on all interfaces	unknown	8016	129	Start	Options
lo		127.0.0.1	0	0	Start	Options

**3.** The 'Start' button for the 'eth1' interface is highlighted.

The status bar at the bottom of the Wireshark window shows: File: "/tmp/etherXXXX1LAUCU" 822 Bytes 0... P: 7 D: 7 M: 0 Drops: 0

# Anwendungsszenarien

The screenshot shows a Debian virtual machine running in Sun xVM VirtualBox. The desktop environment includes a terminal window and a Wireshark window. The terminal window shows the execution of the following commands:

```
debian:/home/linux# wireshark &  
[1] 7512  
debian:/home/linux# ping debian.org  
PING debian.org (192.25.206.10) 56(84) bytes of data.  
[ ]
```

The Wireshark window displays a capture from the eth1 interface. The 'Captured Packets' table is as follows:

Protocol	Total	% of total
Total	32	
SCTP	0	0,0%
TCP	0	0,0%
UDP	0	0,0%
ICMP	31	96,9%
ARP	1	3,1%
OSPF	0	0,0%
GRE	0	0,0%
NetBIOS	0	0,0%
IPX	0	0,0%
VINES	0	0,0%
Other	0	0,0%

The Wireshark window also shows a 'Running' status with a timer at 00:00:31. A red box highlights the 'Stop' button, which is labeled with the number '4.'.

At the bottom of the terminal window, the status bar reads: eth1: <live capture in progress> to file: /t... P: 32

# Anwendungsszenarien

5.

No. .	Time	Source	Destination	Protocol	Info
1	0.000000	CadmusCo_e5:64:92	Broadcast	ARP	who has 10.0.2.3? Tell 10.0.2.15
2	0.000135	RealtekU_12:35:03	CadmusCo_e5:64:92	ARP	10.0.2.3 is at 52:54:00:12:35:03
3	0.000146	10.0.2.15	10.0.2.3	DNS	Standard query A debian.org
4	0.031598	10.0.2.3	10.0.2.15	DNS	Standard query response A 192.25.206.10
5	0.045546	CadmusCo_e5:64:92	Broadcast	ARP	who has 10.0.2.2? Tell 10.0.2.15
6	0.047971	RealtekU_12:35:02	CadmusCo_e5:64:92	ARP	10.0.2.2 is at 52:54:00:12:35:02
7	0.048072	10.0.2.15	192.25.206.10	ICMP	Echo (ping) request
8	1.041949	10.0.2.15	192.25.206.10	ICMP	Echo (ping) request
9	2.041450	10.0.2.15	192.25.206.10	ICMP	Echo (ping) request
10	3.049600	10.0.2.15	192.25.206.10	ICMP	Echo (ping) request
11	4.049483	10.0.2.15	192.25.206.10	ICMP	Echo (ping) request
12	5.054172	10.0.2.15	192.25.206.10	ICMP	Echo (ping) request
13	6.057869	10.0.2.15	192.25.206.10	ICMP	Echo (ping) request
14	7.057834	10.0.2.15	192.25.206.10	ICMP	Echo (ping) request

6.

```
0000 ff ff ff ff ff ff 08 00 27 e5 64 92 08 06 00 01 ..... 'd.....
0010 08 00 06 04 00 01 08 00 27 e5 64 92 0a 00 02 0f ..... 'd.....
```

# Anwendungsszenarien

- Wireshark wird als Superuser gestartet, da der Prozess nicht mit dem Schließen der Konsole beendet werden soll. Wir machen ihn zu einem Hintergrundprozess (1)

```
# wireshark &
```

- um ein wenig Traffic an der Schnittstelle zu erzeugen, pingen wir einfach wahllos einen Host an (1)

```
# ping debian.org
```

- jetzt noch das Interface auswählen, von welchem wir die Pakete mitschneiden wollen, und los gehts (2 & 3)
- Mitschnitt läuft bis wir ihn beenden (4)

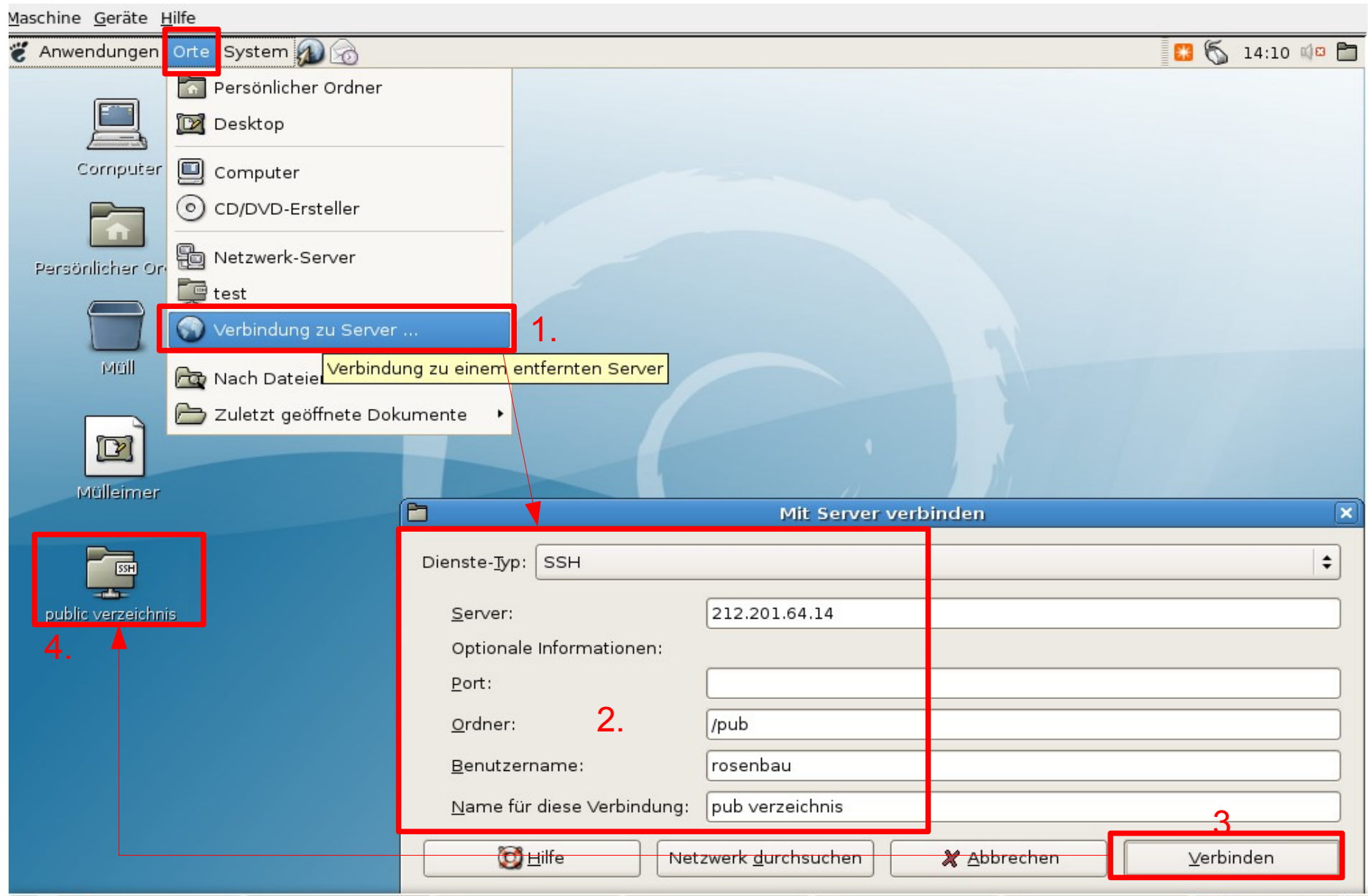
# Anwendungsszenarien

- grafische Ausgabe der mit geschnittenen Pakete (5)
- Paketinhaltsanzeige (6)

# Anwendungsszenarien

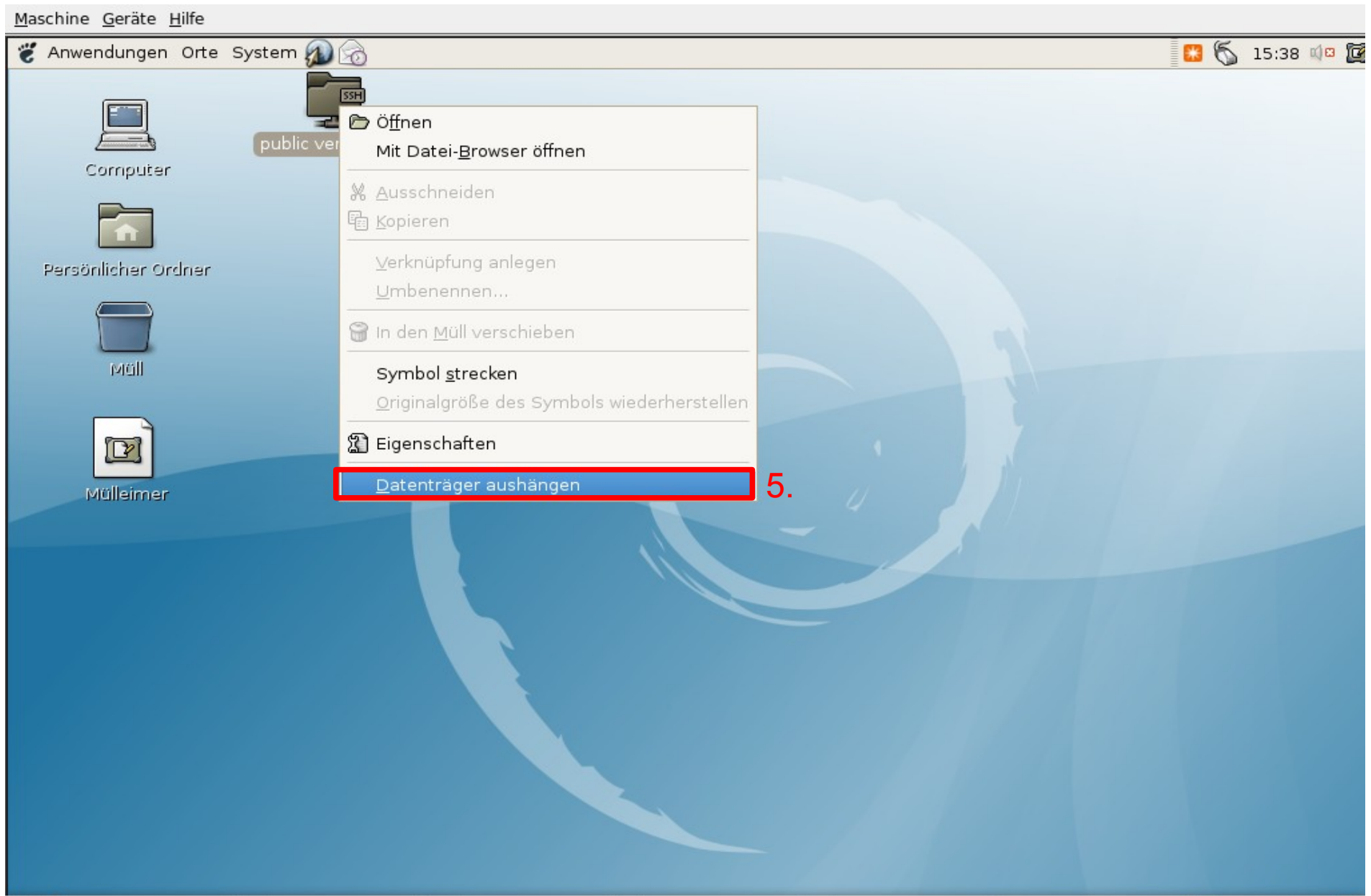
Netzlaufwerk mounten  
(am Beispiel des /pub Verzeichnis)

# Anwendungsszenarien





# Anwendungsszenarien



# Anwendungsszenarien

- unter Gnome ist das einbinden eines Netzlaufwerks recht einfach
- um ein Verknüpfung anzulegen gehen wir über den Reiter „Orte“ und rufen den Eintrag „Verbindung zu Server ...“ auf (1 & 2)
  - Dienst Typ = SSH (SFTP)
  - Server = 212.201.64.14
  - Ordner = /pub
  - Name für diese Verbindung = frei wählbar
- sind alle nötigen Einstellungen getätigt, brauchen wir nur noch „Verbinden“ drücken (3)
- Auf dem Desktop erscheint nun ein Icon mit dem von euch gewählten Namen (4)

# Anwendungsszenarien

- ein klick mit der rechten Maustaste auf das erstellte Icon ermöglicht das aushängen des Laufwerks (5)

# Anwendungsszenarien

FTP

# Anwendungsszenarien

- grafische FTP-Clienten gibt es so einige
- plattformübergreifend sein hier „FileZilla“ genannt, den es für alle gängigen Betriebssystem gibt
- FileZilla wurde in Hinblick auf einfache Bedienung und kostenlose Verfügbarkeit bei gleichzeitiger Unterstützung möglichst vieler nützlicher Features entwickelt
- Arbeiten per Drag & Drop möglich
- FileZilla gehört nicht zu einer Debian Standardinstallation
- <http://www.filezilla.de/download.htm>

# Anwendungsszenarien

The screenshot shows the FileZilla application window. At the top, the title bar reads "FileZilla". Below it is a menu bar with "File", "Edit", "Transfer", "Server", "Help", and "New version available!". A toolbar contains various icons for file operations. The connection settings bar is highlighted with a red box and labeled "1.", showing "Host: 212.201.64", "Username: rosenbau", "Password:" (empty), "Port:" (empty), and a "Quickconnect" dropdown menu. Below this, the command log shows: "Command: PASV", "Response: 227 Entering Passive Mode (212,201,64,14,131,19)", "Command: LIST", "Response: 150 Opening BINARY mode data connection for /bin/ls.", "Response: 226 Transfer complete.", and "Status: Directory listing successful".

The main interface is split into two panes, both highlighted with red boxes and labeled "2." and "3." respectively. The left pane, labeled "2.", shows the "Local site" as "/home/linux/" and a directory tree with folders: bin, boot, cdrom, dev, and etc. Below the tree is a table listing local files and folders.

Filename	Filesize	Filetype	Last modified
..			
.Trash		Folder	Di 17 Jun 20...
.filezilla		Folder	Di 17 Jun 20...
.fontc...		Folder	Do 05 Jun 2...
.gconf		Folder	Di 17 Jun 20...
.gconfd		Folder	Di 17 Jun 20...
.gimp...		Folder	Di 17 Jun 20...
.gno...		Folder	Do 12 Jun 2...

The right pane, labeled "3.", shows the "Remote site" as "/home/rosenbau/". The directory tree shows a "home" folder containing a "rosenbau" folder. Below the tree is a table listing remote files and folders.

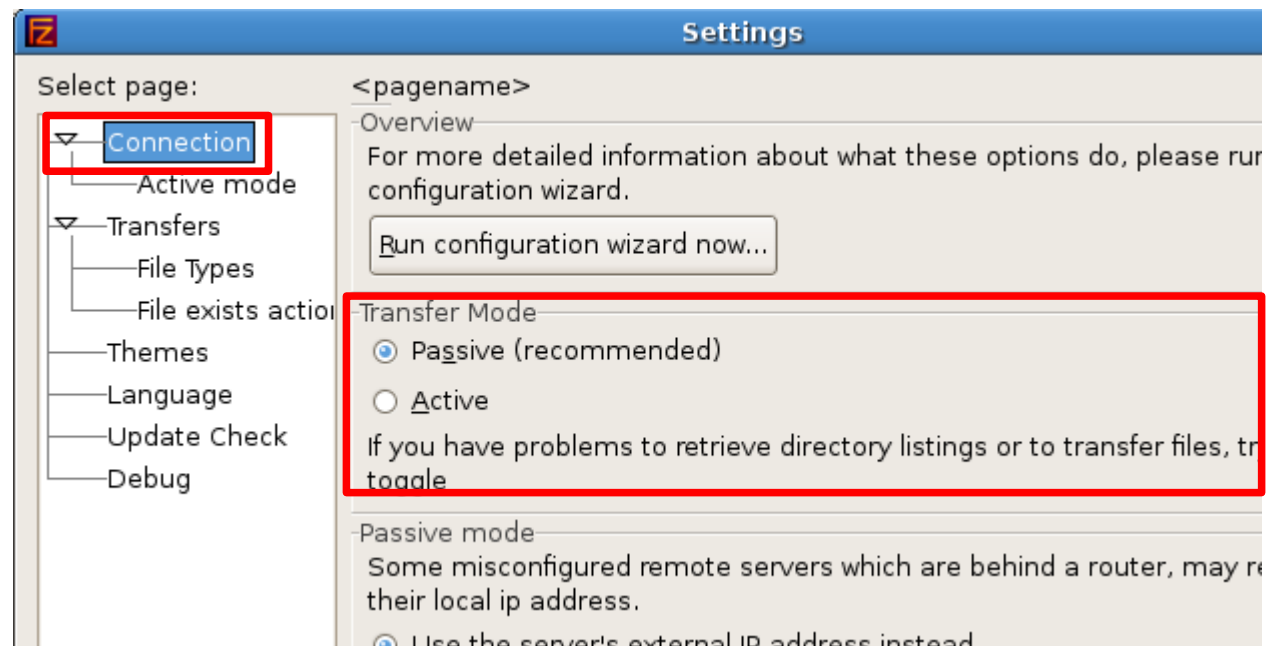
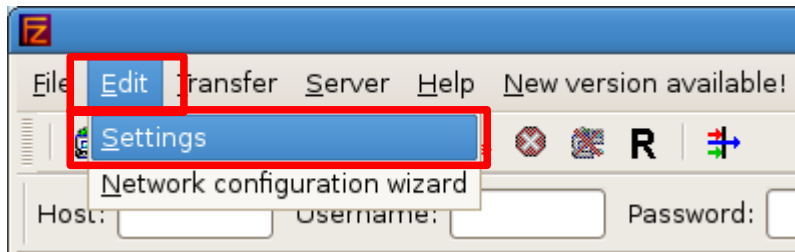
Filename	Filesize	Filetype	Last modified	Permissions	Owner / Group
..					
.adobe		Folder	Mi 28 Mai 20...	drwxr-xr-x	rosenbau.
.gconf		Folder	Mi 28 Mai 20...	drwx-----	rosenbau.
.gconfd		Folder	Mi 28 Mai 20...	drwx-----	rosenbau.
.gnome		Folder	Mi 28 Mai 20...	drwxr-xr-x	rosenbau.
.gno...		Folder	Mi 28 Mai 20...	drwx-----	rosenbau.
.gno...		Folder	Mi 28 Mai 20...	drwx-----	rosenbau.

At the bottom of the window, there are buttons for "Server / Local file", "Directory", "Remote file", "Size", "Priority", and "Status".

# Anwendungsszenarien

- Verbindungsdaten eintragen (1)
- das Verzeichnis auf dem lokalen System (in welchem man sich gerade befindet) mit seiner gesamten Ordnerstruktur wird abgebildet, um so das arbeiten mit Dateien bei einem möglichen Transfer zu erleichtern (2)
- ist man mit einem Server verbunden wird auch dessen Verzeichnisbaum ausgegeben (3)
- Modus (aktiv/passiv) bei Transfer-Problemen wechseln!

# Anwendungsszenarien

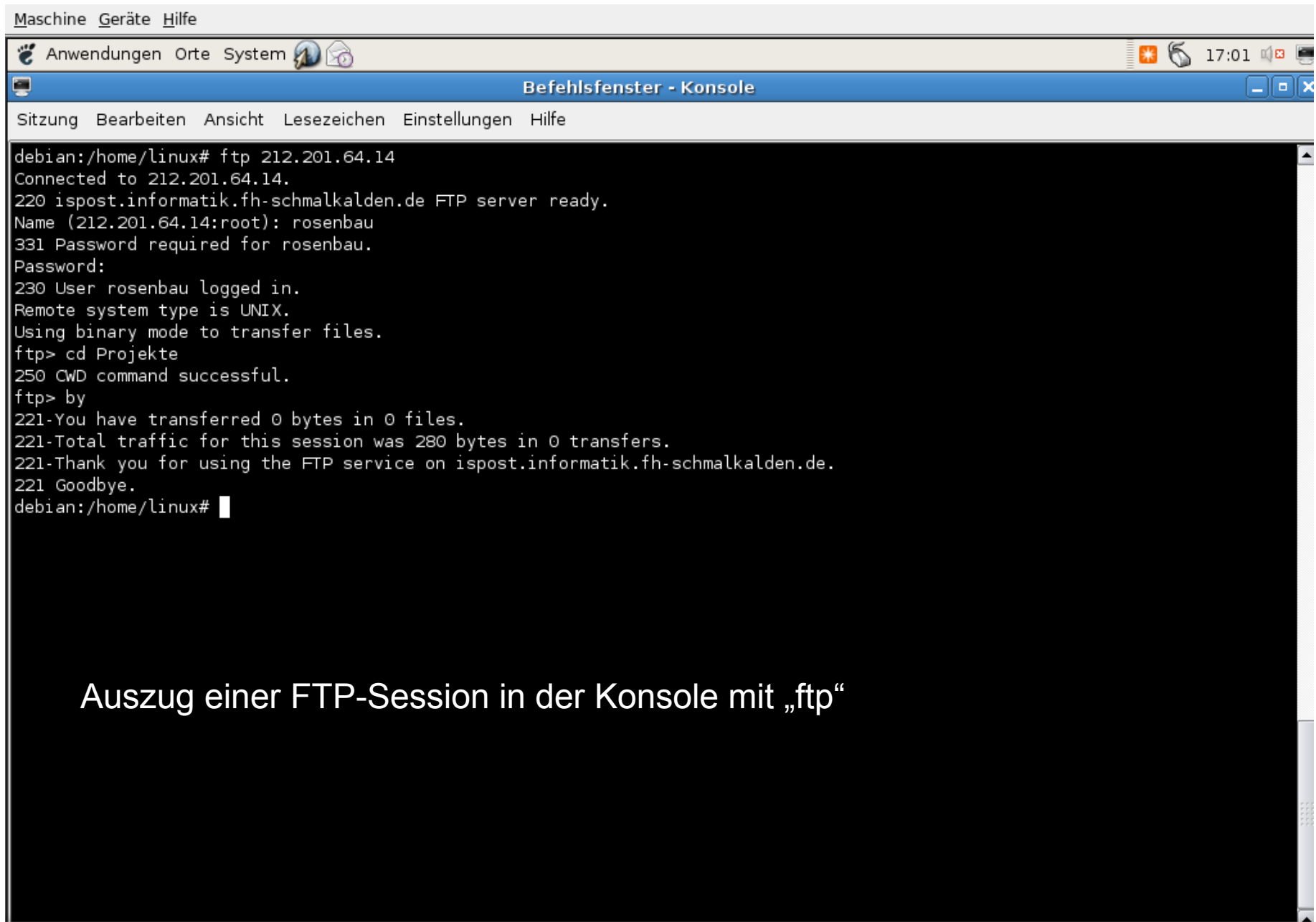




# Anwendungsszenarien

- der Konsolen-Client unter Linux/Unix nennt sich „ftp“ , wie das Protokoll selbst
- startet man ihn, erscheint ein ftp>-Prompt
- in diesem können diverse Befehle ausgeführt werden
  - cd <Verzeichnis> = Wechselt das Arbeitsverzeichnis auf dem Server
  - get <Datei> = Lädt eine Datei herunter
  - put <Datei> = Lädt eine Datei auf den Server.
  - del <Datei> = Löscht eine Datei.
  - ls/dir = zeigt den Verzeichnisinhalt auf dem Server an
  - by/bye = Beendet die aktuelle FTP-Sitzung
  - pwd = Zeigt das Arbeitsverzeichnis auf dem Server an

# Anwendungsszenarien

A screenshot of a Linux terminal window titled "Befehlsfenster - Konsole". The window shows an FTP session initiated from a Debian system. The user enters the command 'ftp 212.201.64.14', connects to the server, logs in as 'rosenbau', and then enters 'by' to exit the session. The terminal output shows the standard FTP protocol messages for connection, login, and disconnection.

```
debian:/home/linux# ftp 212.201.64.14
Connected to 212.201.64.14.
220 ispost.informatik.fh-schmalkalden.de FTP server ready.
Name (212.201.64.14:root): rosenbau
331 Password required for rosenbau.
Password:
230 User rosenbau logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd Projekte
250 CWD command successful.
ftp> by
221-You have transferred 0 bytes in 0 files.
221-Total traffic for this session was 280 bytes in 0 transfers.
221-Thank you for using the FTP service on ispost.informatik.fh-schmalkalden.de.
221 Goodbye.
debian:/home/linux#
```

Auszug einer FTP-Session in der Konsole mit „ftp“

# Anwendungsszenarien

Browser, Messenger & Co.

# Anwendungsszenarien

- Browser
  - eine entsprechende Vielfalt von Browsern steht natürlich auch unter Linux zur Verfügung
  - erhältlich sind u.a. Firefox(Iceweasel bei Debian), Opera, Konqueror (KDE), ...
  - ein echter Linux-User ist natürlich auch in der Lage ohne grafischer Oberfläche im Internet zu surfen
    - Konsolenbrowser „lynx“ und „links“
- Messenger
  - Auch hier ist die Situation vergleichbar mit der Browserwelt

# Anwendungsszenarien

- die meisten Softwarefirmen sind auch im Kommunikationszweig auf den Zug der Plattformunabhängigkeit aufgesprungen
- neben Pidgin (ex Gaim) und Skype gibt es eine Menge alternativen in der Open-Source Gemeinde z.B. Kopete, XChat, Wengophone, ...
- abschließend sei nur gesagt, dass man diese Liste mit Anwendungssoftware beliebig fortsetzen könnte, dies aber den Zeitlichen Rahmen sprengt

Wer sucht, der findet